SCENEPLACER: REFORMULATING SCENE-LEVEL OBJECT POSE ESTIMATION AS VISIBILITY FIELD LEARNING

Anonymous authors
Paper under double-blind review

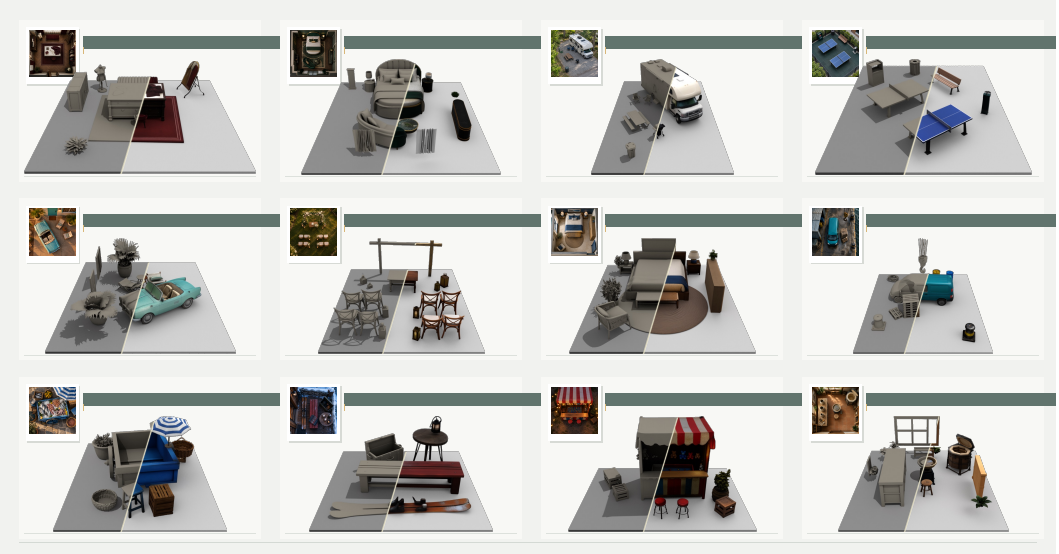

Figure 1: Starting from an input RGB image, and given the generated object and its estimated geometry, our method robustly and efficiently estimates the object's pose within the scene.

ABSTRACT

We present ScenePlacer, a framework for scene-level pose estimation of objects generated by state-of-the-art native shape generation models, placing generated 3D objects into scenes given a partial geometric observation without modifying the underlying generative pipeline. Existing scene generation approaches either generate objects directly in scene coordinates, compromising their canonical-space priors, or regress poses from complete generated shapes in the shape generation process — the latter being a poorly conditioned problem that is sensitive to scale ambiguity and partial observations. ScenePlacer decouples object generation from pose estimation by reformulating alignment as conditional visibility-field learning. Given an observed partial point cloud from reconstruction and a generated mesh, it predicts a visibility field defined in object space, then estimates the rigid transformation by aligning the partial observation to visible parts on the mesh. We propose a pretrained 3D shape VAE as our backbone, preserving its mesh prior while conditioning it on a separately encoded observation. We further introduce visibility-aware, radius-constrained supervision and synthesize training pairs using only object-level meshes and rendered views, eliminating the need for scene-level pose annotations. On 3D-FUTURE, ScenePlacer consistently outperforms existing placement and pose-refinement baselines under both ground-truth and monocular depth estimation. With generated objects, it achieves a Chamfer distance of 0.0077 and an F-score of 92.40 under estimated depth and improves to 0.0023 and 98.27, respectively, under ground-truth geometry.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

Compositional 3D scene generation is essential for AR/VR and embodied AI. Recent 3D object generative models provide strong priors for recovering complete geometry from sparse observations, which can be transferred to this task. Thanks to the capabilities of 3D generative models (Zhang et al., 2023; Xiang et al., 2025), scene quality has improved significantly compared to before, yet there is still a considerable gap before the results are truly satisfactory, coming with degraded shape quality, excessive memory consumption, and inaccurate pose estimation.

Several methods generate objects within the scene in place. One strategy (Huang et al., 2025) generates all objects in a shared canonical space, inevitably degraded due to limited resolution. The other strategy (Siddiqui et al., 2026; Yin et al., 2026) tries to inject the observed partial point cloud as a condition into the 3D generative model and generates each object sequentially. However, the observed point clouds are incomplete, and their orientation varies with the camera viewpoint. By requiring the model to generate objects with arbitrary translations and rotations, this approach violates the common assumption in 3D generative models that objects are centered at the origin and aligned with the coordinate axes, inevitably degrading generation quality. An alternative approach (Meng et al., 2026; Chen et al., 2026) preserves the canonical generation space and introduces a separate pose prediction head that estimates the transformation (translation, rotation, and scale) from object to the world coordinate frame. Although this maintains high-quality object generation, pose prediction accuracy remains a major concern: Small perturbations in the transformation matrix can lead to large geometric deviations once applied, rendering the regression problem ill-conditioned.

Instead of struggling balancing the generation quality and pose estimation accuracy like existing methods, we view placing generated objects back into the scene as a distinct problem that warrants dedicated investigation. This problem presents challenges beyond conventional pose estimation. A generated object may preserve the structural or functional characteristics of the observed object without faithfully reproducing its visible geometry, making rigid registration unreliable. Moreover, generated objects are typically normalized to $[-1, 1]^3$ and lack scene-scale information, introducing scale ambiguity that further complicates rendering-based pose optimization. Any6D (Lee et al., 2025) addresses this problem through multi-stage pose and scale refinement, but relies on close geometric agreement between the generated object and the observation, with errors potentially propagating and accumulating across refinement stages. We propose a dedicated single-pass pose alignment model that operates on the generated mesh, enabling efficient and robust pose estimation without affecting generation quality. Specifically, given an observed partial point cloud $\mathcal{P}_{\text{obs}}$ of an object from reconstruction and its corresponding generated mesh $\mathcal{M}_{\text{gen}}$, we propose to predict the mesh-aligned visible surface $\mathcal{P}_{\text{aligned}}$ and solve for the transformation between $\mathcal{P}_{\text{obs}}$ and $\mathcal{P}_{\text{aligned}}$.

Our method is based on the insight that pose estimation can be solved by finding the visible surface on a mesh in object canonical space corresponding to the observed part. This reframing removes the dependency on scene-level data, which remains scarce in practice. Architecturally, we observe that our task—predicting a partial surface aligned with a given complete mesh—naturally aligns with the prior embedded in 3D Shape VAE, which encodes and decodes complete shapes. By treating the observed partial point cloud as a conditioning signal that modulates the mesh decoder, our network only needs to learn to select the visible subset of the mesh surface. This design enables strong generalization with only lightweight tuning. Our approach is designed to be general-purpose and independent of the underlying generation pipeline. It does not require modifications to the generated object or retraining of the generator, enabling straightforward integration with diverse object generators and geometry models.

In summary, our contributions are as follows:

- We recast object estimation as learning a *visibility field* with respect to the complete object mesh, transforming the pose estimation problem into a conditional surface prediction task that is inherently object-centric and independent of scene-level context.
- We introduce a lightweight pose estimation module that can be seamlessly integrated with existing generative models. It predicts the mesh-aligned visible surface and resolves the rigid transformation between the observed partial point cloud and the generated object, without altering the generation process.

- We propose a self-supervised training paradigm that relies solely on object-level data, repurposing a 3D shape VAE to address object placement in compositional scene generation, thereby eliminating the need for scarce scene-level supervision.

2 Related Work

Geometry Foundation Models. Recent geometry foundation models adopt transformer architecture for traditional 3D vision tasks such as depth estimation, camera pose estimation, and 3D point tracking. Some of them focus on single-view depth estimation (Yang et al., 2024a;b; Wang et al., 2025b; 2026b; Xu et al., 2026; Viola et al., 2025), aiming to learn RGB-to-Depth transformation, while they can't handle multi-view input. Dust3R (Wang et al., 2024) tried to handle stereo reconstruction by formulating the problem as direct regression of a point map for the second view in the first view's camera coordinate system. Moving forward, VGGT and a series of subsequent works (Wang et al., 2025a; Deng et al., 2026a; Wang et al., 2026d; Lin et al., 2025) tried to extend to multi-view input: VGGT directly predicts per-view depth maps and per-view camera poses; DA3 adopts ray map as camera representation; Pi3 focuses on learning pose-invariant 3D representations that are agnostic to camera motion; Sail-Recon extends VGGT to long sequence input. However, the above works can only handle static scenes. Recent geometry foundation model D4RT (Zhang et al., 2026) focuses on 4D scene reconstruction.

Scene Generation Scene generation approaches fall into two categories. The first leverages 3D object generation priors (Hunyuan3D et al., 2025; Zhang et al., 2024; Li et al., 2024; 2025b; Xiang et al., 2025; Deng et al., 2026b; Ye et al., 2025; Luo et al., 2026; Yan et al., 2026; Li et al., 2026b; Wu et al., 2026; Xiang et al., 2026), together with geometry foundation and pattern recognition priors (Deng et al., 2026c; Carion et al., 2026; Ren et al., 2024), and extends them to decomposable scene generation (Ardelean et al., 2025; Huang et al., 2025; Chen et al., 2026; Shi et al., 2026; Yin et al., 2026; Yao et al., 2025; Meng et al., 2026; Siddiqui et al., 2026); in such scenes, objects are interactive and compatible with physics simulation. Some (Feng et al., 2023; Yang et al., 2024c; Bian et al., 2025; Hao et al., 2026; Wang et al., 2025c; Pfaff et al., 2026) use language models to orchestrate scenes from text prompts, producing more orderly layouts but lacking realism and image alignment. The second utilizes 2D image or video priors and incorporates 3D conditions (e.g., camera parameters) to generate multi-view image or video sequences with 3D consistency (Huang et al., 2026; Fang et al., 2025; 2026; Liang et al., 2025; Yu et al., 2025; Szymanowicz et al., 2025; Li et al., 2025a; Gao et al., 2024; Sun et al., 2026; Wang et al., 2026c; Shen et al., 2026). While these methods achieve superior visual quality via 2D priors, lacking native 3D representation makes ensuring 3D interaction and physical consistency difficult.

Point Cloud Registration Point cloud registration is closely related to single-reference unseen object pose estimation, since both tasks depend on establishing reliable correspondences between partial 3D observations. Classical approaches typically operate on hand-crafted geometric cues. For instance, ICP and its variants (Besl & McKay, 1992; Rusinkiewicz & Levoy, 2001) alternate between nearest-neighbor association and rigid transform refinement, while FPFH (Rusu et al., 2009) characterizes local surface geometry via point feature histograms, and PPF (Drost et al., 2010) represents oriented point-pair relations for generating pose hypotheses. Although these methods serve as reasonable geometric baselines, they are often fragile under outliers, repetitive structures, and limited view overlap. Learning-based techniques attempt to overcome these limitations by predicting more distinctive correspondence signals. Predator (Huang et al., 2021), for example, introduces overlap-aware feature learning so that matching concentrates on regions that are actually shared between two partially overlapping scans. Similarly, GeoTransformer (Qin et al., 2023) aggregates pairwise distances and angular information into rotation-invariant geometric features, thereby enabling robust superpoint matching across large viewpoint changes. More recently, Any6D (Lee et al., 2025) addresses this setting in a model-free manner by reconstructing a normalized object shape from a single RGB-D anchor image, jointly aligning it to metric scale in 2D/3D, and selecting render-and-compare pose hypotheses for the query image.

3 Preliminary: 3DShape2VecSet Representation

We build on the 3DShape2VecSet Zhang et al. (2023) representation, which encodes a 3D shape as a set of latent vectors $\{\mathbf{f}_i\}_{i=1}^{M}$ without explicit spatial coordinates. Given an input point cloud with

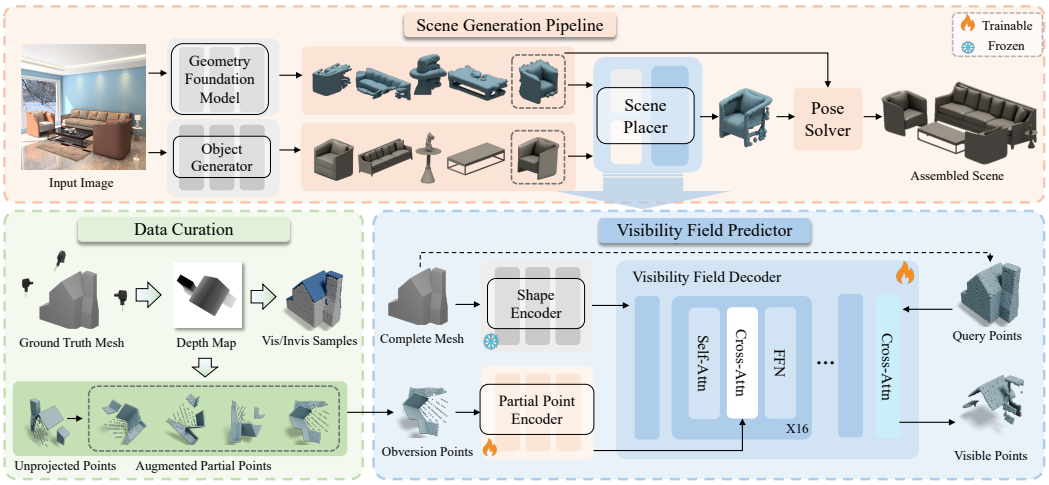

Figure 2: **Overview.** *Pipeline:* The input image is first processed by a geometry foundation model to obtain a point cloud and a mesh generated by a generative model; ScenePlacer is then used to place the object back into its corresponding spatial location, resulting in a compositional scene. *Data Curation:* Depth maps rendered from the ground-truth mesh are back-projected into oriented partial points, augmented by random translation, rotation and scales, and vis/invis samples with an exclusion radius provide supervision. *Visibility Field Predictor:* A frozen Shape Encoder and a trainable Partial Point Encoder produce mesh and observation latents, respectively; the Visibility Field Decoder conditions the mesh latents on the observation, and decodes visibility for query points sampled on the mesh surface, yielding the visible points used for alignment.

coordinates $\mathbf{X}$ and normals $\mathbf{N}$, the encoder uses cross-attention to aggregate per-point features into M latents:

$$\{\mathbf{f}_i\} = \mathrm{CrossAttn}\big(\mathrm{PosEmb}(\mathbf{X}_0, \mathbf{N}_0),\ \mathrm{PosEmb}(\mathbf{X}, \mathbf{N})\big), \tag{1}$$

where $\mathbf{X}_0$ is obtained by farthest point sampling from $\mathbf{X}$, and $\mathbf{N}_0$ denotes the normals at the sampled points. For decoding, an occupancy value at query coordinate $\mathbf{x}$ is predicted by attending from $\mathbf{x}$ to the latent set:

$$\hat{\mathcal{O}}(\mathbf{x}) = \mathrm{Linear}\left(\sum_i \mathbf{v}(\mathbf{f}_i) \cdot \mathrm{softmax}\big(\mathbf{q}(\mathbf{x})^\top \mathbf{k}(\mathbf{f}_i)\big)\right). \tag{2}$$

This defines a continuous occupancy field that can be queried at arbitrary points.

4 Method

Let $\mathcal{M}$ denote the object mesh and $\mathcal{P}_{\text{obs}}$ an observed partial point cloud reconstructed from image(s) or scanning. The observed cloud is initially unaligned with the mesh and contains only 3D coordinates. We reformulate the alignment task as *visibility field learning*: a network F takes the mesh and the observation as input and predicts, for any query point $\mathbf{x} \in \mathbb{R}^3$, the probability that $\mathbf{x}$ lies on the visible surface of $\mathcal{M}$ as seen from the observation viewpoint. Formally,

$$F(\mathcal{M}, \mathcal{P}_{\text{obs}}) : \mathbb{R}^3 \to [0,1], \quad \hat{\mathcal{V}}(\mathbf{x}) = F(\mathcal{M}, \mathcal{P}_{\text{obs}})(\mathbf{x}), \tag{3}$$

where $\hat{\mathcal{V}}(\mathbf{x})$ denotes the predicted visibility. The aligned visible surface $\mathcal{P}_{\text{aligned}}$ is obtained by sampling points with $\hat{\mathcal{V}}(\mathbf{x}) > \tau$. The rigid transformation between the mesh and the observed cloud is then recovered by aligning the two partial point sets:

$$\mathcal{T}^* = \arg\min_{\mathcal{T}} \mathrm{CD}\big(\mathcal{T}(\mathcal{P}_{\text{aligned}}),\ \mathcal{P}_{\text{obs}}\big), \tag{4}$$

where $\mathcal{T}$ denotes a rigid transformation and CD denotes the chamfer distance between two point sets. The network is trained with binary cross-entropy supervision: positive queries are sampled from the rendered visible surface, while negative queries are drawn from the mesh surface

with an exclusion radius that ensures they correspond to genuinely occluded or unobserved regions (Sec. 4.3). Throughout this pipeline, point clouds serve as the sole geometric representation. The overview of our method can be found in Figure 2.

4.1 Observation Latent Encoding

To encode the observed point cloud and the complete point cloud into a shared latent space, we also represent the encoding of the observed point cloud in the form of a Vecset and initialize both encoders using the same weights. We encode the partial observation and the complete mesh with separate encoders: the mesh encoder is frozen from a pretrained encoder, while the observation encoder adopts the same architecture but is fully trainable, preventing the partial observation from disturbing the mesh prior. The observed point cloud $\mathcal{P}_{\text{obs}}$ contains only 3D coordinates and no normals. We first downsample it to a subset $\mathcal{P}_0$ via farthest point sampling, then use cross-attention to aggregate positional embeddings from both sets into M observation latents:

$$\{\mathbf{o}_i\} = \text{CrossAttn}\big(\text{PosEmb}(\mathcal{P}_0),\ \text{PosEmb}(\mathcal{P}_{\text{obs}})\big). \tag{5}$$

In contrast to the original 3DShape2VecSet formulation, we omit the KL regularization bottleneck. Since our goal is conditional reconstruction rather than generative modeling, retaining the full latent capacity helps preserve fine geometric detail and reduces the risk of entangling observation noise with structural information in a compressed latent space.

4.2 Visibility Field Predictor

The mesh decoder is originally trained to reconstruct complete, watertight surfaces from mesh latents alone, and thus provides a strong geometric prior. We do not discard this prior but instead use the observation latents to condition it. Specifically, we interleave cross-attention layers into the decoder's self-attention stack. Let $\mathbf{M} = \{\mathbf{m}_i\}_{i=1}^{M}$ denote the mesh latents and $\mathbf{O} = \{\mathbf{o}_i\}_{i=1}^{M}$ the observation latents. Each decoder block is defined as:

$$\mathbf{M}^{(l+1)} = \text{CrossAttn}\big(\text{SelfAttn}(\mathbf{M}^{(l)}),\ \mathbf{O}\big), \tag{6}$$

where the mesh latents act as queries and the observation latents provide keys and values. This asymmetric design allows the mesh latents to retain their original representational capacity while being modulated by observed evidence. The decoder therefore does not regress the visible surface from scratch; instead, it biases its existing mesh reconstruction toward the region that is consistent with the observation. To ensure stable initialization, the final projection is zero-initialized, so that at the start of training the conditioning path introduces no distortion and the model initially behaves as the unconditional mesh decoder.

After obtaining the modulated mesh latents $\mathbf{M}^{(L)}$, we decode visibility by sampling query points directly on the mesh surface. For each query point $\mathbf{x} \in \mathcal{S}_{\mathcal{M}}$, we compute

$$\hat{v}(\mathbf{x}) = \sigma\Big(\text{Linear}\Big(\sum_i \mathbf{v}(\mathbf{m}_i^{(L)}) \cdot \text{softmax}\Big(\mathbf{q}(\mathbf{x})^\top \mathbf{k}(\mathbf{m}_i^{(L)})\Big)\Big)\Big), \tag{7}$$

where σ is the sigmoid function and $\mathbf{q}, \mathbf{k}, \mathbf{v}$ are learnable projections. The predicted visible surface is obtained by thresholding:

$$\mathcal{P}_{\text{aligned}} = \{\mathbf{x} \in \mathcal{S}_{\mathcal{M}} \mid \hat{v}(\mathbf{x}) > 0.5\}. \tag{8}$$

4.3 Visibility-Aware Supervision

The visibility field is supervised in a self-supervised manner. Visible queries are sampled directly from the canonical visible surface $\mathcal{P}_{\text{aligned}}$ and assigned label $y = 1$. Invisible queries are sampled from the full mesh surface, but are accepted only if they lie outside an exclusion radius r_{excl} from any visible point. Formally, let $\mathcal{R}$ be a reference subset of $\mathcal{P}_{\text{aligned}}$. A candidate invisible point p_{inv} is accepted only if

$$\min_{r \in \mathcal{R}} \|p_{\text{inv}} - r\|_2 > r_{\text{excl}}. \tag{9}$$

This constraint is critical for learning a correct visible/invisible separation. If invisible queries were sampled uniformly from the mesh surface, many would fall near the boundary of the visible region or in the gaps between visible surface patches. These points are geometrically close to visible points but

semantically belong to the occluded or unobserved part of the object. Without the exclusion radius, the network receives contradictory supervision and tends to interpolate between visible and invisible regions instead of learning a sharp visibility boundary. By enforcing a hard radius constraint around visible points, we ensure that invisible queries correspond unambiguously to non-visible surface regions, which stabilizes training and improves the quality of the predicted visible surface.

Let $\mathcal{Q} = \mathcal{Q}^+ \cup \mathcal{Q}^-$ denote the set of all accepted query points. The network is trained with a binary cross-entropy loss:

$$\mathcal{L}_{\text{BCE}} = -\frac{1}{|\mathcal{Q}|} \sum_{\mathbf{x} \in \mathcal{Q}} \Big[y(\mathbf{x}) \log \hat{v}(\mathbf{x}) + \big(1 - y(\mathbf{x})\big) \log \big(1 - \hat{v}(\mathbf{x})\big) \Big], \tag{10}$$

where $y(\mathbf{x}) = 1$ for $\mathbf{x} \in \mathcal{Q}^+$ and $y(\mathbf{x}) = 0$ for $\mathbf{x} \in \mathcal{Q}^-$.After training, the predicted visible surface is used to recover the relative transformation by minimizing the chamfer distance.

4.4 Training Data Synthesis

Training requires paired examples of observed partial point clouds and their corresponding mesh-aligned visible surfaces. For each object, we pre-render depth maps from 60 uniformly sampled viewpoints to ensure coverage of diverse observation conditions. At each training iteration, we randomly select 1–3 viewpoints per object and back-project the depth maps into the mesh coordinate frame using the known camera poses. This yields a mesh-aligned partial point cloud $\mathcal{P}_{\text{aligned}}$.

To synthesize the network input, we apply a random similarity transformation to $\mathcal{P}_{\text{aligned}}$:

$$\hat{\mathcal{P}}_{\text{obs}} = \{sRp + t \mid p \in \mathcal{P}_{\text{aligned}}\}, \tag{11}$$

where $s \in \mathbb{R}^+$, $R \in SO(3)$, and $t \in \mathbb{R}^3$ are randomly sampled, followed by normalization to a unit sphere. We refer to $\hat{\mathcal{P}}_{\text{obs}}$ as a training proxy for real observations. Since the ground-truth alignment is known by construction, the network can be trained in a self-supervised manner to recover $\mathcal{P}_{\text{aligned}}$ from $\hat{\mathcal{P}}_{\text{obs}}$ conditioned on the mesh.

5 Experiments

5.1 Experimental Setup

Datasets and Benchmarks. We evaluate our method on 3D-Future (Fu et al., 2021), a scene dataset composed of numerous object layouts that can be decomposed into individual meshes. For training, we select 130,000 high-quality meshes from Objaverse (Deitke et al., 2023), training with learning rate of 1e-4 for 30000 steps. We evaluate on a randomly sampled subset of 1,000 scenes from 3D-FUTURE test set. Ablation study are conducted on 250 scenes randomly drawn from this subset.

Baselines. We compare our method against several paradigms for scene-level layout prediction. **3DFixer** performs in-place generation conditioned on a reconstructed point cloud of the target object, directly producing a surface conforming to the observed geometry. **SAM3D** generates objects in canonical space and employs a layout denoising network to predict translation, rotation, and scale in the world coordinate frame. **SAM3D+PO** (Pose Optimization) initializes from SAM3D predictions and refines the pose by optimizing rendering over the mask and and perform rigid registration between the observed point cloud and the complete point cloud. This refinement aligns the *complete* generated mesh against the *observed partial* point cloud. **Any6D** is a recent 6D pose estimation framework and we utilize the 3D objects generated by SAM3D as the reference shapes and apply the alignment scheme from Any6D to align the predicted depth and objects for scene-level layout prediction. In contrast, our pose estimation operates directly between two partial point sets without relying on rendering loss and SAM3D initialization: the network-predicted visible surface and the observed partial cloud.

Metrics. We jointly assess reconstruction fidelity and layout accuracy using Chamfer Distance (CD) and F-score at $\tau = 0.05$ and $\tau = 0.10$, computed over concatenated object-surface point clouds ($N = 4096$ per object via triangle-area-weighted sampling). To remove global unit scale while preserving relative placement, we apply a single isotropic normalization $x' = (x - c_{\text{GT}}) \cdot 1.9/s_{\text{GT}}$

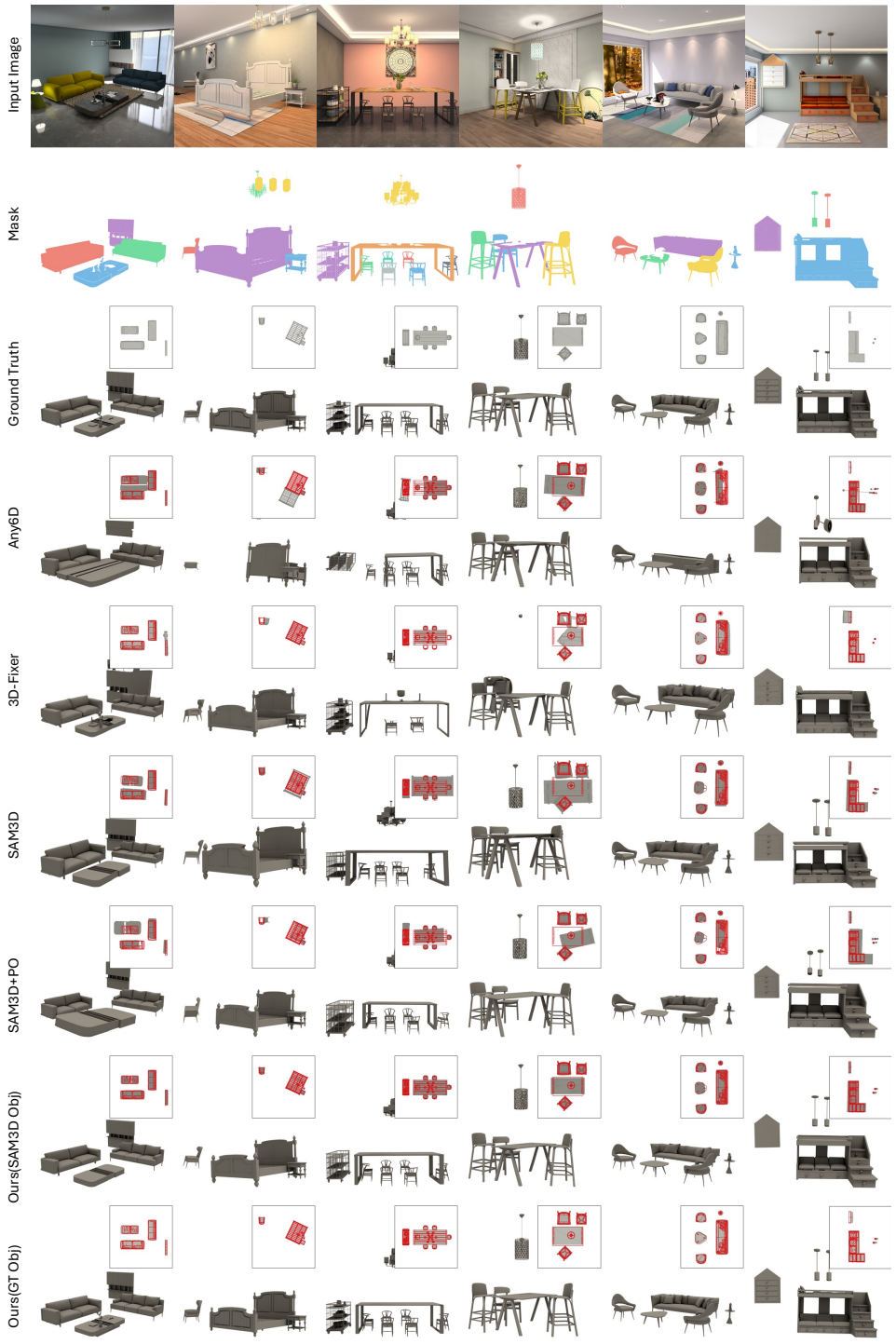

Figure 3: Qualitative comparison of scene-level layout predictions (The rightmost column uses MoGe Depth, while the others use GT-Depth.). Each row shows a scene generated by different methods, with the ground-truth layout shown for reference (red outline). Our method produces layouts that closely align with the ground truth, while baselines exhibit noticeable errors.

derived from the ground-truth axis-aligned bounding box to both clouds, without any subsequent registration (no ICP, rotation fitting, or per-object alignment). Translation, rotation, scale, and object-layout errors therefore remain fully observable in the reported metrics. For ease of reading, the metrics Chamfer Distance (CD) and F-score mentioned below are reported as values scaled by 100.

Table 1: Scene-level quantitative comparison under different depth inputs. All values are multiplied by 100. Best in bold, second best underlined.

Depth	Metric	Any6D	3DFixer	SAM3D	SAM3D+PO	Ours (SAM3D obj.)	Ours (GT obj.)
GT	CD ↓	1.04	23.35	0.98	1.51	<u>0.36</u>	**0.23**
	F@0.05 ↑	85.18	85.65	75.01	74.26	<u>90.38</u>	**95.60**
	F@0.10 ↑	92.81	89.63	90.73	87.05	<u>97.01</u>	**98.27**
MoGe	CD ↓	2.87	23.43	1.31	1.92	<u>0.77</u>	**0.70**
	F@0.05 ↑	74.14	76.51	71.40	65.79	<u>78.53</u>	**81.94**
	F@0.10 ↑	86.80	86.48	88.77	83.48	<u>92.40</u>	**93.38**

5.2 COMPARISON

Table 1 reports scene-level results under ground-truth depth and MoGe-predicted depth. Across all metrics and settings, our method consistently outperforms all baselines, as illustrated in Figure 3. More results can be found in supplementary material.

Ground-truth depth. We use ground-truth geometry as input, which rules out the effect of imperfect geometry estimation, allowing us to observe and compare the output quality of different methods. 3D-Fixer struggles to consistently produce high-quality geometry, and may even fail to generate under conditions such as complex occlusion (i.e., no complete mesh can be extracted), with a CD of 23.35—over an order of magnitude worse than ours. SAM3D generates reasonably but has limited layout accuracy, with F@0.05 of 75.01 versus our 90.38. SAM3D+PO degrades performance versus SAM3D alone. We attribute this to two factors. First, rendering-based alignment is inherently scale-ambiguous: a larger, farther object and a smaller, nearer object can project to nearly identical rendered images, so rendered loss cannot constrain the true scale. Second, its geometric loss aligns a complete mesh to a partial observation; mesh regions absent from the observed point cloud introduce a systematic bias, preventing convergence to correct surface correspondences. Any6D, aligning SAM3D-generated objects with observed depth via its pose and size estimation, achieves a competitive F@0.05 of 85.18, outperforming SAM3D. However, its CD remains high at 1.04, indicating its mechanism still struggles to recover fine-grained geometric details.

In stark contrast, our method achieves exceptional reconstruction fidelity and layout accuracy, driving the CD down to an impressive 0.36 and pushing the F@0.10 to 97.01. This remarkable performance stems from the core strength of our conditional visible-surface reconstruction paradigm. By directly predicting the mesh-aligned visible surface, our network establishes highly precise, dense partial-to-partial correspondences with the observed point cloud. Consequently, under the perfect geometric guidance of GT depth, our method perfectly anchors the generated objects into the scene, demonstrating that our partial-to-partial alignment mechanism is theoretically sound and practically optimal for leveraging high-quality depth observations.

MoGe-predicted depth. Under estimated depth input, the performance gap widens further. 3DFixer remains severely degraded, while Any6D suffers from a significant performance drop, with its CD soaring to 2.87 and F@0.05 dropping to 74.14. This highlights that the alignment scheme in Any6D is highly sensitive to depth noise, likely due to scale ambiguity and the lack of robust partial-to-partial matching under noisy observations. SAM3D+PO again fails to improve upon its initialization, consistent with the inherent limitations of full-to-partial alignment under noisy depth. With SAM3D-generated objects, our method achieves a CD of 0.77, a 41% relative improvement over SAM3D and a 60% improvement over SAM3D+PO. Notably, our training does not incorporate MoGe-predicted depth for data augmentation, unlike 3DFixer and SAM3D. The robust performance under noisy depth therefore stems from the inherent resilience of our conditional visible-surface reconstruction paradigm rather than from domain-specific augmentation.

Analysis with upstream quality. To decouple our core alignment module from upstream generation artifacts, we evaluate our method under a perfect setting (i.e., using GT depth and GT objects). This configuration establishes the representational upper bound of our framework, purely reflecting the geometric representation capacity and optimization efficacy of our conditional visible-surface reconstruction. Under these idealized geometric conditions, our method achieves exceptional precision, with CD dropping to 0.23 and F@0.10 soaring to 98.27%. These near-perfect metrics validate the efficacy of our visibility-aware supervision, demonstrating that our network possesses the capacity to establish highly accurate partial-to-partial correspondences when freed from upstream noise. Our method already attains state-of-the-art placement accuracy with current upstream outputs, while the consistent gains observed when using higher-fidelity geometry demonstrate that our framework can readily benefit from further advances in 3D generation and reconstruction.

Table 2: Ablation study. All values are multiplied by 100.

Depth	Metric	Shared-Encoder	Attn-Swap (w/o Aug.)	Ours (w/o Vis. RCS)	Ours (full)
GT	CD ↓	0.57	1.90	0.52	**0.34**
	F@0.10 ↑	93.86	81.75	96.65	**96.69**
MoGe	CD ↓	0.99	2.07	0.87	**0.74**
	F@0.10 ↑	88.53	76.75	91.84	**92.14**

5.3 Ablation Study

We conduct ablation experiments to isolate the contribution of each key design choice. Results under both depth settings are reported in Table 2.

The necessity of separating encoding of mesh and observation. We compare against a **Shared-Encoder** variant that uses one trainable encoder for both the complete mesh and the observed partial cloud, instead of a frozen mesh encoder and a separate observation encoder. As shown in Table 2, it degrades under both settings: GT CD 0.34→0.57 and F@0.10 96.69→93.86; MoGe CD 0.74→0.99 and F@0.10 92.14→88.53. This confirms that separate encoding is essential to prevent the partial, noisy observation from disturbing the well-learned mesh prior.

Attention Swap. Reversing the attention direction (**Attn-Swap**), where the observation latents serve as queries and the mesh latents provide context, results in a dramatic performance drop: CD increases from 0.349 to 1.905 under GT depth and from 0.787 to 2.071 under MoGe depth. This confirms that retaining the mesh as the primary representation—with the observation acting as a modulator—is essential for preserving the strong geometric prior learned by the mesh decoder.

The necessity of Radius-Constrained Sampling. Removing the exclusion radius (**Ours (w/o Vis. RCS)**) degrades performance, with CD increasing from 0.349 to 0.520 under GT depth and from 0.787 to 0.869 under MoGe depth. This occurs because negative samples crowd near the visible/invisible boundary without a radius constraint, creating ambiguous supervision. Introducing a moderate exclusion radius establishes a clear buffer zone that explicitly separates visible and invisible regions, providing sharp supervision that stabilizes training and improves the fidelity of the predicted visible surface.

6 Conclusion

We introduce ScenePlacer, a scene-level object pose estimation method that formulates the observation-to-generation correspondence by learning a visibility field. Given a generated mesh and an observed partial point cloud, ScenePlacer predicts a visibility field defined in object space, extracts the visible surface on the generated mesh, and recovers the pose through partial-to-generation alignment. The method leverages a pretrained 3D shape representation, 3DShape2VecSet, uses visibility-aware supervision, and can be trained solely on object-level data without scene-level pose annotations. Experiments on 3D-FUTURE demonstrate consistent improvements in geometric fidelity and layout accuracy over existing baselines under both ground-truth and estimated depth, while the framework remains compatible with future advances in 3D generation and monocular geometry estimation.

REFERENCES

Andreea Ardelean, Mert Özer, and Bernhard Egger. Gen3dsr: Generalizable 3d scene reconstruction via divide and conquer from a single view. In *2025 International Conference on 3D Vision (3DV)*, pp. 616–626. IEEE, 2025.

Paul J Besl and Neil D McKay. Method for registration of 3-d shapes. In *Sensor fusion IV: control paradigms and data structures*, volume 1611, pp. 586–606. Spie, 1992.

Zixuan Bian, Ruohan Ren, Yue Yang, and Chris Callison-Burch. Holodeck 2.0: Vision-language-guided 3d world generation with editing. *arXiv preprint arXiv:2508.05899*, 2025.

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris Coll-Vinent, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, et al. Sam 3: Segment anything with concepts. In *International conference on learning representations*, volume 2026, pp. 138846–138923, 2026.

Xingyu Chen, Fu-Jen Chu, Pierre Gleize, Kevin J Liang, Alexander Sax, Hao Tang, Weiyao Wang, Michelle Guo, Thibaut Hardin, Xiang Li, et al. Sam 3d: 3dfy anything in images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7220–7232, 2026.

Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 13142–13153, 2023.

Junyuan Deng, Heng Li, Tao Xie, Weiqiang Ren, Qian Zhang, Ping Tan, and Xiaoyang Guo. Sailrecon: Large sfm by augmenting scene regression with localization. In *2026 International Conference on 3D Vision (3DV)*, pp. 309–329. IEEE, 2026a.

Ken Deng, Yuan-Chen Guo, Jingxiang Sun, Zi-Xin Zou, Yangguang Li, Xin Cai, Yan-Pei Cao, Yebin Liu, and Ding Liang. Detailgen3d: Generative 3d geometry enhancement via data-dependent flow. In *2026 International Conference on 3D Vision (3DV)*, pp. 1–31. IEEE, 2026b.

Ken Deng, Yunhan Yang, Jingxiang Sun, Xihui Liu, Yebin Liu, Ding Liang, and Yan-Pei Cao. Geosam2: Unleashing the power of sam2 for 3d part segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6367–6376, 2026c.

Bertram Drost, Markus Ulrich, Nassir Navab, and Slobodan Ilic. Model globally, match locally: Efficient and robust 3d object recognition. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pp. 998–1005. Ieee, 2010.

Chuan Fang, Yuan Dong, Kunming Luo, Xiaotao Hu, Rakesh Shrestha, and Ping Tan. Ctrl-room: Controllable text-to-3d room meshes generation with layout constraints. In *2025 International Conference on 3D Vision (3DV)*, pp. 692–701. IEEE, 2025.

Chuan Fang, Lingteng Qiu, Yixun Liang, Rui Chen, Kunming Luo, Zhaohua Zheng, Tongyuan Bai, Feipeng Tian, Zilong Dong, Zihan Zhou, et al. Spatialcrafter: Single image world modeling with generative 3d proxies. *arXiv preprint arXiv:2608.27073*, 2026.

Weixi Feng, Wanrong Zhu, Tsu-jui Fu, Varun Jampani, Arjun Akula, Xuehai He, Sugato Basu, Xin Eric Wang, and William Yang Wang. Layoutgpt: Compositional visual planning and generation with large language models. *Advances in neural information processing systems*, 36: 18225–18250, 2023.

Huan Fu, Rongfei Jia, Lin Gao, Mingming Gong, Binqiang Zhao, Steve Maybank, and Dacheng Tao. 3d-future: 3d furniture shape with texture. *International Journal of Computer Vision*, 129 (12):3313–3337, 2021.

Ruiqi Gao, Aleksander Holynski, Philipp Henzler, Arthur Brussee, Ricardo Martin-Brualla, Pratul Srinivasan, Jonathan T Barron, and Ben Poole. Cat3d: Create anything in 3d with multi-view diffusion models. *arXiv preprint arXiv:2405.10314*, 2024.

Jinkun Hao, Naifu Liang, Zhen Luo, Xudong Xu, Weipeng Zhong, Ran Yi, Yichen Jin, Zhaoyang Lyu, Feng Zheng, Lizhuang Ma, et al. Mesatask: Towards task-driven tabletop scene generation via 3d spatial reasoning. *Advances in neural information processing systems*, 38:122057–122099, 2026.

Jiaxin Huang, Yuanbo Yang, Bangbang Yang, Lin Ma, Yuewen Ma, and Yiyi Liao. Gen3r: 3d scene generation meets feed-forward reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 25358–25369, 2026.

Shengyu Huang, Zan Gojcic, Mikhail Usvyatsov, Andreas Wieser, and Konrad Schindler. Predator: Registration of 3d point clouds with low overlap. In *2021 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 4265–4274. IEEE, 2021.

Zehuan Huang, Yuan-Chen Guo, Xingqiao An, Yunhan Yang, Yangguang Li, Zi-Xin Zou, Ding Liang, Xihui Liu, Yan-Pei Cao, and Lu Sheng. Midi: Multi-instance diffusion for single image to 3d scene generation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23646–23657. IEEE, 2025.

Team Hunyuan3D, Shuhui Yang, Mingxin Yang, Yifei Feng, Xin Huang, Sheng Zhang, Zebin He, Di Luo, Haolin Liu, Yunfei Zhao, Qingxiang Lin, Zeqiang Lai, Xianghui Yang, Huiwen Shi, Zibo Zhao, Bowen Zhang, Hongyu Yan, Lifu Wang, Sicong Liu, Jihong Zhang, Meng Chen, Liang Dong, Yiwen Jia, Yulin Cai, Jiaao Yu, Yixuan Tang, Dongyuan Guo, Junlin Yu, Hao Zhang, Zheng Ye, Peng He, Runzhou Wu, Shida Wei, Chao Zhang, Yonghao Tan, Yifu Sun, Lin Niu, Shirui Huang, Bojian Zheng, Shu Liu, Shilin Chen, Xiang Yuan, Xiaofeng Yang, Kai Liu, Jianchen Zhu, Peng Chen, Tian Liu, Di Wang, Yuhong Liu, Linus, Jie Jiang, Jingwei Huang, and Chunchao Guo. Hunyuan3d 2.1: From images to high-fidelity 3d assets with production-ready pbr material, 2025. URL https://arxiv.org/abs/2506.15442.

Taeyeop Lee, Bowen Wen, Minjun Kang, Gyuree Kang, In So Kweon, and Kuk-Jin Yoon. Any6d: Model-free 6d pose estimation of novel objects, 2025. URL https://arxiv.org/abs/2503.18673.

Baicheng Li, Dong Wu, Jun Li, Shunkai Zhou, Zecui Zeng, Lusong Li, and Hongbin Zha. Mv-sam3d: Adaptive multi-view fusion for layout-aware 3d generation. *arXiv preprint arXiv:2603.11633*, 2026a.

Weiyu Li, Jiarui Liu, Hongyu Yan, Rui Chen, Yixun Liang, Xuelin Chen, Ping Tan, and Xiaoxiao Long. Craftsman3d: High-fidelity mesh generation with 3d native generation and interactive geometry refiner. *arXiv preprint arXiv:2405.14979*, 2024.

Xinyang Li, Tengfei Wang, Zixiao Gu, Shengchuan Zhang, Chunchao Guo, and Liujuan Cao. Flashworld: High-quality 3d scene generation within seconds. In *The Fourteenth International Conference on Learning Representations*, 2025a.

Yangguang Li, Zi-Xin Zou, Zexiang Liu, Dehu Wang, Yuan Liang, Zhipeng Yu, Xingchao Liu, Yuan-Chen Guo, Ding Liang, Wanli Ouyang, et al. Triposg: High-fidelity 3d shape synthesis using large-scale rectified flow models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025b.

Zhihao Li, Yufei Wang, Heliang Zheng, Yihao Luo, and Bihan Wen. Sparc3d: Sparse representation and construction for high-resolution 3d shapes modeling. *Advances in Neural Information Processing Systems*, 38:118582–118600, 2026b.

Hanwen Liang, Junli Cao, Vidit Goel, Guocheng Qian, Sergei Korolev, Demetri Terzopoulos, Konstantinos N Plataniotis, Sergey Tulyakov, and Jian Ren. Wonderland: Navigating 3d scenes from a single image. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 798–810. IEEE, 2025.

Haotong Lin, Sili Chen, Junhao Liew, Donny Y Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. *arXiv preprint arXiv:2511.10647*, 2025.

Kunming Luo, Hongyu Yan, Yuan Liu, Zihao Zhang, Manyuan Zhang, Wenping Wang, and Ping Tan. Ctr3d: Cross-view token reduction for dense multi-view generation. In *2026 International Conference on 3D Vision (3DV)*, pp. 956–966, 2026. doi: 10.1109/3DV69130.2026.00096.

Yanxu Meng, Haoning Wu, Ya Zhang, and Weidi Xie. Scenegen: Single-image 3d scene generation in one feedforward pass. In *2026 International Conference on 3D Vision (3DV)*, pp. 543–553. IEEE, 2026.

Nicholas Pfaff, Thomas Cohn, Sergey Zakharov, Rick Cory, and Russ Tedrake. Scenesmith: Agentic generation of simulation-ready indoor scenes. In *Forty-third International Conference on Machine Learning*, 2026.

Zheng Qin, Hao Yu, Changjian Wang, Yulan Guo, Yuxing Peng, Slobodan Ilic, Dewen Hu, and Kai Xu. Geotransformer: Fast and robust point cloud registration with geometric transformer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(8):9806–9821, 2023.

Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, et al. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159*, 2024.

Szymon Rusinkiewicz and Marc Levoy. Efficient variants of the icp algorithm. In *Proceedings third international conference on 3-D digital imaging and modeling*, pp. 145–152. IEEE, 2001.

Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. Fast point feature histograms (fpfh) for 3d registration. In *2009 IEEE international conference on robotics and automation*, pp. 3212–3217. IEEE, 2009.

Tianchang Shen, Sherwin Bahmani, Kai He, Sangeetha Grama Srinivasan, Tianshi Cao, Jiawei Ren, Ruilong Li, Zian Wang, Nicholas Sharp, Zan Gojcic, et al. Lyra 2.0: Explorable generative 3d worlds. *arXiv preprint arXiv:2604.13036*, 2026.

Yukai Shi, Weiyu Li, Zihao Wang, Hongyang Li, Xingyu Chen, Ping Tan, and Lei Zhang. Scenemaker: Open-set 3d scene generation with decoupled de-occlusion and pose estimation model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 27146–27156, 2026.

Yawar Siddiqui, Duncan Frost, Samir Aroudj, Armen Avetisyan, Henry Howard-Jenkins, Daniel DeTone, Pierre Moulon, Qirui Wu, Zhengqin Li, Julian Straub, et al. Shaper: Robust conditional 3d shape generation from casual captures. *arXiv preprint arXiv:2601.11514*, 2026.

Yang-Tian Sun, Zehuan Huang, Yifan Niu, Lin Ma, Yan-Pei Cao, Yuewen Ma, and Xiaojuan Qi. Stereo world model: Camera-guided stereo video generation. *arXiv preprint arXiv:2603.17375*, 2026.

Stanislaw Szymanowicz, Jason Y Zhang, Pratul Srinivasan, Ruiqi Gao, Arthur Brussee, Aleksander Holynski, Ricardo Martin-Brualla, Jonathan T Barron, and Philipp Henzler. Bolt3d: Generating 3d scenes in seconds. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 24846–24857. IEEE, 2025.

Massimiliano Viola, Kevin Qu, Nando Metzger, Bingxin Ke, Alexander Becker, Konrad Schindler, and Anton Obukhov. Marigold-dc: Zero-shot monocular depth completion with guided diffusion. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5359–5370. IEEE, 2025.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5294–5306. IEEE, 2025a.

Jianyuan Wang, Minghao Chen, Shangzhan Zhang, Nikita Karaev, Johannes Schönberger, Patrick Labatut, Piotr Bojanowski, David Novotny, Andrea Vedaldi, and Christian Rupprecht. Vggt-$Omega$. *arXiv preprint arXiv:2605.15195*, 2026a.

Ruicheng Wang, Sicheng Xu, Cassie Dai, Jianfeng Xiang, Yu Deng, Xin Tong, and Jiaolong Yang. Moge: Unlocking accurate monocular geometry estimation for open-domain images with optimal training supervision. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5261–5271. IEEE, 2025b.

Ruicheng Wang, Sicheng Xu, Yue Dong, Yu Deng, Jianfeng Xiang, Zelong Lv, Guangzhong Sun, Xin Tong, and Jiaolong Yang. Moge-2: Accurate monocular geometry with metric scale and sharp details. *Advances in Neural Information Processing Systems*, 38:35928–35959, 2026b.

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20697–20709. IEEE, 2024.

Weijie Wang, Xiaoxuan He, Youping Gu, Yifan Yang, Zeyu Zhang, Yefei He, Yanbo Ding, Xirui Hu, Donny Y Chen, Zhiyuan He, et al. World-r1: Reinforcing 3d constraints for text-to-video generation. *arXiv preprint arXiv:2604.24764*, 2026c.

Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. π^3: Permutation-equivariant visual geometry learning, 2026d. URL https://arxiv.org/abs/2507.13347.

Ziqian Wang, Yonghao He, Licheng Yang, Wei Zou, Hongxuan Ma, Liu Liu, Wei Sui, Yuxin Guo, and Hu Su. Tabletopgen: Instance-level interactive 3d tabletop scene generation from text or single image. *arXiv e-prints*, pp. arXiv–2512, 2025c.

Shuang Wu, Youtian Lin, Feihu Zhang, Yifei Zeng, Yikang Yang, Jiachen Qian, Siyu Zhu, Xun Cao, Philip Torr, Yao Yao, et al. Direct3d-s2: Gigascale 3d generation made easy with spatial sparse attention. *Advances in Neural Information Processing Systems*, 38:170778–170804, 2026.

Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3d latents for scalable and versatile 3d generation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21469–21480. IEEE, 2025.

Jianfeng Xiang, Xiaoxue Chen, Sicheng Xu, Ruicheng Wang, Zelong Lv, Yu Deng, Hongyuan Zhu, Yue Dong, Hao Zhao, Nicholas Jing Yuan, and Jiaolong Yang. Native and compact structured latents for 3d generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14419–14429, June 2026.

Gangwei Xu, Haotong Lin, Hongcheng Luo, Xianqi Wang, Jingfeng Yao, Lianghui Zhu, Yuechuan Pu, Cheng Chi_, Haiyang Sun, Bing Wang, et al. Pixel-perfect depth with semantics-prompted diffusion transformers. *Advances in Neural Information Processing Systems*, 38:174731–174755, 2026.

Hongyu Yan, Kunming Luo, Weiyu Li, Kaiyi Zhang, Yixun Liang, Jingwei Huang, Chunchao Guo, and Ping Tan. Posemaster: A unified 3d native framework for stylized pose generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 34292–34302, 2026.

Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10371–10381. IEEE, 2024a.

Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *Advances in neural information processing systems*, 37:21875–21911, 2024b.

Yue Yang, Fan-Yun Sun, Luca Weihs, Eli VanderBilt, Alvaro Herrasti, Winson Han, Jiajun Wu, Nick Haber, Ranjay Krishna, Lingjie Liu, et al. Holodeck: Language guided generation of 3d embodied ai environments. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16277–16287. IEEE, 2024c.

Kaixin Yao, Longwen Zhang, Xinhao Yan, Yan Zeng, Qixuan Zhang, Lan Xu, Wei Yang, Jiayuan Gu, and Jingyi Yu. Cast: Component-aligned 3d scene reconstruction from an rgb image. *ACM Transactions on Graphics (TOG)*, 44(4):1–19, 2025.

Chongjie Ye, Yushuang Wu, Ziteng Lu, Jiahao Chang, Xiaoyang Guo, Jiaqing Zhou, Hao Zhao, and Xiaoguang Han. Hi3dgen: High-fidelity 3d geometry generation from images via normal bridging. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 01–12. IEEE, 2025.

Ze-Xin Yin, Liu Liu, Xinjie Wang, Wei Sui, Zhizhong Su, Jian Yang, and Jin Xie. 3d-fixer: Coarse-to-fine in-place completion for 3d scenes from a single image. *arXiv preprint arXiv:2604.04406*, 2026.

Hong-Xing Yu, Haoyi Duan, Charles Herrmann, William T Freeman, and Jiajun Wu. Wonderworld: Interactive 3d scene generation from a single image. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5916–5926. IEEE, 2025.

Biao Zhang, Jiapeng Tang, Matthias Nießner, and Peter Wonka. 3dshape2vecset: A 3d shape representation for neural fields and generative diffusion models. *ACM Trans. Graph.*, 42(4), jul 2023. ISSN 0730-0301. doi: 10.1145/3592442. URL https://doi.org/10.1145/3592442.

Chuhan Zhang, Guillaume Le Moing, Skanda Koppula, Ignacio Rocco, Liliane Momeni, Junyu Xie, Shuyang Sun, Rahul Sukthankar, Joëlle K. Barral, Raia Hadsell, Zoubin Ghahramani, Andrew Zisserman, Junlin Zhang, and Mehdi S. M. Sajjadi. Efficiently reconstructing dynamic scenes one d4rt at a time. In *CVPR*, 2026.

Longwen Zhang, Ziyu Wang, Qixuan Zhang, Qiwei Qiu, Anqi Pang, Haoran Jiang, Wei Yang, Lan Xu, and Jingyi Yu. Clay: A controllable large-scale generative model for creating high-quality 3d assets. *ACM Transactions On Graphics (TOG)*, 43(4):1–20, 2024.

A APPENDIX

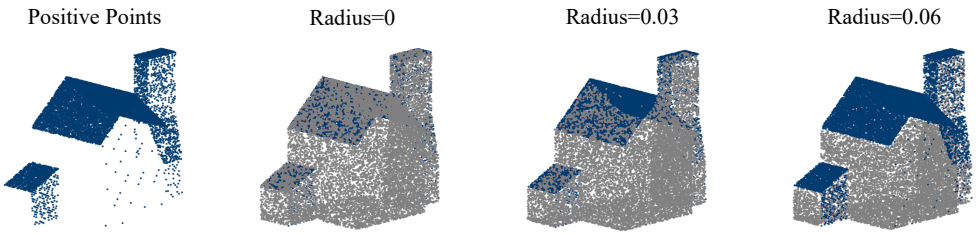

Figure 4: Visualization of positive (visible, blue) and negative (invisible, gray) point sampling under different radius.



Figure 5: Visibility field visualization.

A.1 THE USE OF LARGE LANGUAGE MODELS (LLMS)

We use Large Language Models (LLMs) strictly for minor language editing—grammar and readability, not for method design or experiments. All technical contributions, including the methodology, equations, and results, are solely the work of the authors.

A.2 MORE IMPLEMENTATION DETAILS

We use the pretrained Hunyuan3D-2.1 ShapeVAE (Hunyuan3D et al., 2025) as a point-cloud shape encoder. The original encoder is configured with a 1024-dimensional transformer width, 16 attention heads, 8 encoder layers, Fourier positional features with 8 frequencies, and a 64-dimensional latent space. The pretrained model uses 4096 latent queries and an input surface size of 81,920 points; for efficiency, we reduce the number of latent queries to 512 while retaining the full input surface, resulting in an effective downsampling ratio of 160. The full-shape encoder is frozen and used to provide canonical shape tokens. A second encoder with the same initialization is used for partial

observations. In the main experiments, the pre-KL encoder tokens are used as partial-shape context. The resulting Hunyuan tokens are injected into the occupancy decoder through partial-condition cross-attention.

A.3 More Ablation Study

Radius Constraint Discussion. Incorporating a radius constraint mechanism has significantly improved the sampling of invisible points used for supervision. We visualized the distribution of sampled invisible points based on different radius settings, as shown in Figure 4: if the radius is too small, the invisible points intermingle with visible points—making it difficult to form a continuous surface and causing the field to fragment under supervision—whereas an excessively large radius results in a lack of supervision at the edges. We further visualized the constructed visibility field on the mesh, where the pink areas represent visibility and the white areas represent invisibility; it can be seen that our visibility field effectively captures the relationship between visible and invisible regions formed on the surface through depth back-projection, as illustrated in Figure 5.

Table 3: Ablation study. All values are multiplied by 100.

Depth	Metric	Only Pose Solver	Ours
GT	CD ↓	1.32	**0.34**
	F@0.10 ↑	86.02	**96.69**
MoGe	CD ↓	1.71	**0.74**
	F@0.10 ↑	81.34	**92.14**

Only Pose Solver. We have already conducted a similar experiment by comparing our method with SAM3D with pose optimization. The pose optimization first preprocesses the generated mesh and checks mask reliability, then performs a heuristic alignment using scale and translation estimated from the point map. ICP then refines the alignment, and finally a render-and-compare stage optimizes the pose via silhouette matching.

To rule out other influencing factors, we conduct an ablation study that uses only a pose solver to align the mesh with the observed partial point cloud. As shown in Table 3, direct alignment without visibility field prediction leads to severe degradation, further demonstrating the effectiveness of our method.

A.4 Limitations and Future Work

Despite our placement method's strong performance, its accuracy is bounded by the quality of observed partial points and generated object meshes. Feed-forward RGB depth estimation still trails traditional geometry reconstruction, and image-conditioned object generation, though silhouette-consistent, lacks robustness to multi-view inputs. These are fundamental challenges for the 3D community, and improving underlying geometry foundation models is essential. Future work will focus on enhancing generation methods' robustness to diverse inputs and improving the geometric priors our placement module builds upon.

A.5 More Visualization Results

Since our method is trained on object-level data, it is insensitive to scene content; it not only applies to indoor scenes but also generalizes to outdoor scenes (Figure 6), and even extends to cartoon-style environments (Figure 7).

Moreover, our method can generalize to multi-view inputs. We estimate the geometry via VGGT-Omega (Wang et al., 2026a) and generate the object via MV-SAM3D (Li et al., 2026a). However, existing generative models struggle to directly handle multi-view inputs, so we rely on successfully generated cases to demonstrate the generalization ability of our method under multi-view inputs, as shown in Figure 8. We further tested more complex scenarios; since our method places each object individually, it is not sensitive to scene complexity, and we continue to demonstrate excellent generalization capabilities in more complex indoor and outdoor environments and more visualization

results can be found in Figure 9. We encourage readers to view our accompanying video, which shows the rotation of scenes rendered.

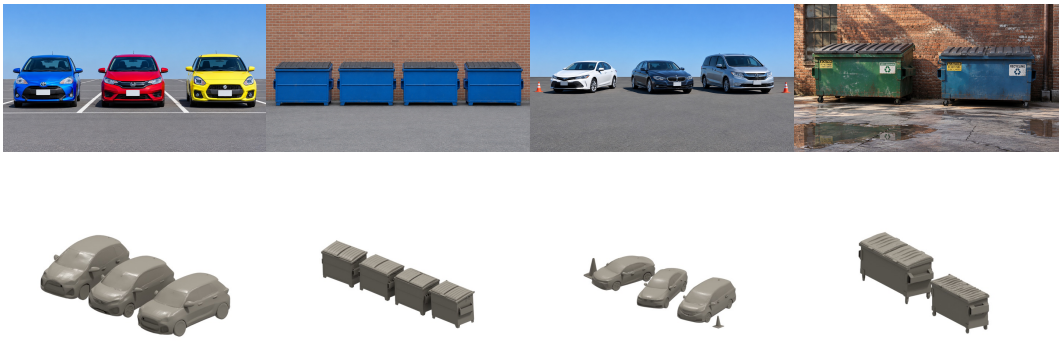

Figure 6: Generalize to outdoor scenes.

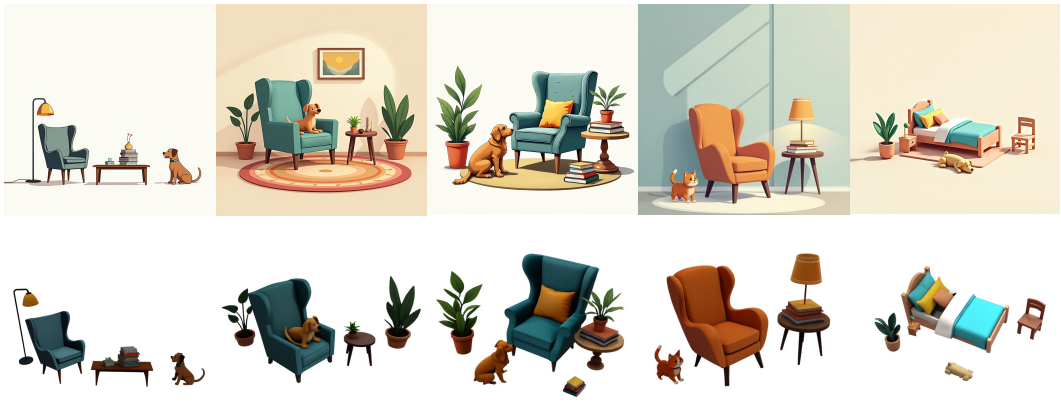

Figure 7: Generalize to cartoon style.



Figure 8: Generalize to multi-view input.

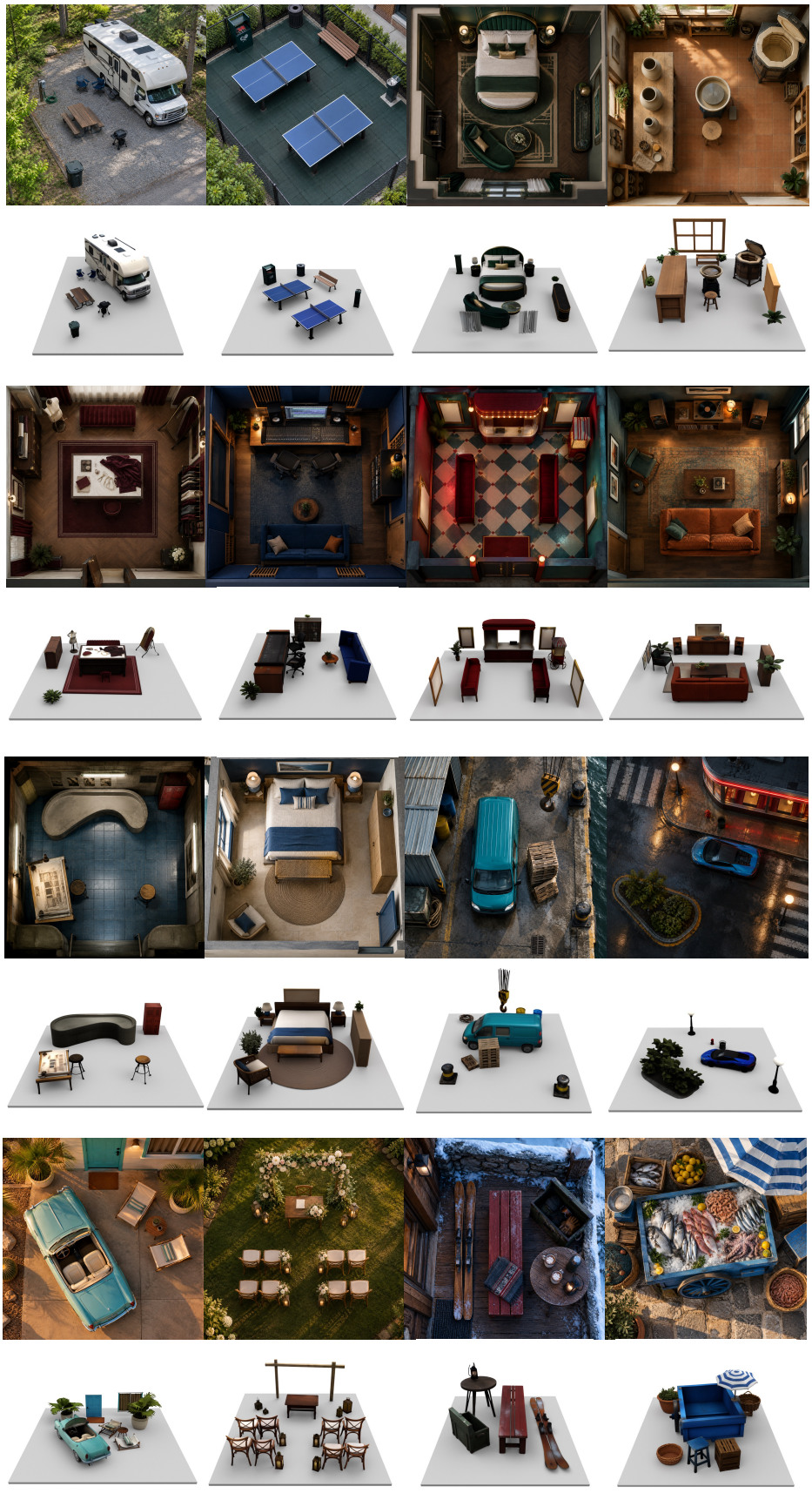

Figure 9: More visualization results.